

WHAT IS CLAIMED IS:

1. A method of distributing processing loads, comprising:
 - (a) receiving a first data packet;
 - (b) determining whether or not any packet processors have been previously selected to process said first data packet based on a classification of said first data packet;
 - 5 (c) when none of said packet processors has been previously designated to process said first data packet, selecting a first selected processor of said packet processors to process said first data packet, wherein said first selected processor is selected based on processing load values respectively corresponding to processing loads of said packet processors.
- 10 2. The method as claimed in claim 1, wherein said classification of said first data packet corresponds to a process flow to which said first data packet belongs.
3. The method as claimed in claim 2, wherein said step (c) comprises:
 - (c1) selecting said first selected processor to process all data packets corresponding to said process flow of said first data packet.
4. The method as claimed in claim 1, wherein said step (c) comprises:
 - (c1) determining which packet processor of said packet processors has a lowest processing load based on said processing load values; and

5 (c2) selecting said packet processor having said lowest processing load as said first selected processor.

5. The method as claimed in claim 1, wherein each of said processing load values represents one of a plurality of processing load categories,

wherein said processing load categories range from a smallest 5 processing load category to a largest processing load category.

6. The method as claimed in claim 5, wherein said step (c) comprises:

(c1) when none of said packet processors have been previously designated to process said first data packet, selecting said first selected 5 processor to process said first data packet based on which of said packet processors has a processing load value corresponding to a lowest processing load category of said processing load categories.

7. The method as claimed in claim 1, wherein step (c) comprises:

(c1) when none of said packet processors has been previously designated to process said first data packet, using a controller to select said first selected processor of said packet processors to process said first data 5 packet, wherein said controller selects said first selected processor based on said processing load values respectively corresponding to said processing loads of said packet processors.

8. The method as claimed in claim 7, wherein said processing load values are respectively generated in said packet processors and respectively output to said controller.

9. The method as claimed in claim 7, wherein said controller determines said processing load values of said packet processors based on a percentage of a total number of tasks that said controller has assigned to each of said packet processors.

10. The method as claimed in claim 9, wherein said tasks correspond to process flows of data packets that said controller has designated said packet processors to process.

11. The method as claimed in claim 7, wherein said processing load values correspond to an idle time of each of said packet processors.

12. The method as claimed in claim 7, wherein said processing load values correspond to a rate at which data is supplied to each of said packet processors.

13. A software program contained on a computer readable medium, wherein the software program enables a controller to perform a routine, comprising:

- (a) receiving a first data packet;
- 5 (b) determining whether or not any packet processors have been previously selected to process said first data packet based on a classification of said first data packet;

10 (c) when none of said packet processors has been previously designated to process said first data packet, selecting a first selected processor of said packet processors to process said first data packet, wherein said first selected processor is selected based on processing load values respectively corresponding to processing loads of said packet processors.

14. The software program as claimed in claim 13, wherein said classification of said first data packet corresponds to a process flow to which said first data packet belongs.

15. The software program as claimed in claim 14, wherein said step (c) comprises:

(c1) selecting said first selected processor to process all data packets corresponding to said process flow of said first data packet.

16. The software program as claimed in claim 13, wherein said step (c) comprises:

(c1) determining which packet processor of said packet processors has a lowest processing load based on said processing load values; and

5 (c2) selecting said packet processor having said lowest processing load as said first selected processor.

17. The software program as claimed in claim 13, wherein each of said processing load values represents one of a plurality of processing load categories,

5 wherein said processing load categories range from a smallest processing load category to a largest processing load category.

18. The software program as claimed in claim 17, wherein said step (c) comprises:

(c1) when none of said packet processors have been previously designated to process said first data packet, selecting said first selected processor to process said first data packet based on which of said packet processors has a processing load value corresponding to a lowest processing load category of said processing load categories.

19. The software program as claimed in claim 13, wherein said processing load values are respectively generated in said packet processors and respectively output to said controller.

20. The software program as claimed in claim 13, wherein said controller determines said processing load values of said packet processors based on a percentage of a total number of tasks that said controller has assigned to each of said packet processors.

21. The software program as claimed in claim 20, wherein said tasks correspond to process flows of data packets that said controller has designated said packet processors to process.

22. The software program as claimed in claim 13, wherein said processing load values correspond to an idle time of each of said packet processors.

23. The software program as claimed in claim 13, wherein said processing load values correspond to a rate at which data is supplied to each of said packet processors.

24. An apparatus for distributing processing loads, comprising:

a controller; and

a plurality of packet processors coupled to said controller,

wherein said controller receives a first data packet and determines

5 whether or not any of said packet processors have been previously selected to process said first data packet based on a classification of said first data packet,

wherein, when none of said packet processors has been previously designated to process said first data packet, said controller selects a first selected processor of said packet processors to process said first data packet,

10 wherein said first selected processor is selected based on processing load values respectively corresponding to processing loads of said packet processors.

25. The apparatus as claimed in claim 24, wherein said classification of said first data packet corresponds to a process flow to which said first data packet belongs.

26. The apparatus as claimed in claim 25, wherein said controller selects said first selected processor to process all data packets corresponding to said process flow of said first data packet.

27. The apparatus as claimed in claim 24, wherein said controller determines which packet processor of said packet processors has a lowest processing load based on said processing load values, and

5 wherein said controller selects said packet processor having said lowest processing load as said first selected processor.

28. The apparatus as claimed in claim 24, wherein each of said processing load values represents one of a plurality of processing load categories,

wherein said processing load categories range from a smallest 5 processing load category to a largest processing load category.

29. The apparatus as claimed in claim 28, wherein, when none of said packet processors have been previously designated to process said first data packet, said controller selects said first selected processor to process said first data packet based on which of said packet processors has a processing 5 load value corresponding to a lowest processing load category of said processing load categories.

30. The apparatus as claimed in claim 24, wherein said processing load values are respectively generated in said packet processors and respectively output to said controller.

31. The apparatus as claimed in claim 24, wherein said controller determines said processing load values of said packet processors based on a percentage of a total number of tasks that said controller has assigned to each of said packet processors.

32. The apparatus as claimed in claim 31, wherein said tasks correspond to process flows of data packets that said controller has designated said packet processors to process.

33. The apparatus as claimed in claim 24, wherein said processing load values correspond to an idle time of each of said packet processors.

34. The apparatus as claimed in claim 24, wherein said processing load values correspond to a rate at which data is supplied to each of said packet processors.

35. A method for distributing processing loads of a plurality of packet processors, comprising:

- (a) receiving input data packets from a stream of data packets;
- (b) determining which of said input data packets are unclassified data packets, wherein said unclassified data packets have unclassified data packet classifications and none of said packet processors has been previously designated to process data packets having any of said unclassified data packet classifications; and
- (c) selecting designated processors of said packet processors to process said unclassified data packets based on load information, wherein said packet processors respectively have processing loads and said load information corresponds to said processing loads.

36. The method as claimed in claim 35, wherein step (b) comprises:

- (b1) determining which of said input data packets are classified data packets, wherein said classified data packets have classified data packet classifications and one of said packet processors has been previously designated to process data packets having said classified data packet classifications.

37. The method as claimed in claim 36, wherein step (c) comprises:

(c1) supplying said classified data packets to said packet processors based on which of said packet processors have been previously designated to process data packets having said classified data packet classifications.

38. The method as claimed in claim 35, wherein said unclassified data packet classifications correspond to process flows of said unclassified data packets.

39. The method as claimed in claim 37, wherein said unclassified data packet classifications correspond to process flows of said unclassified data packets and wherein said classified data packet classifications correspond to process flows of said classified data packets.

40. The method as claimed in claim 35, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

5 wherein each of said processing load values categorizes a corresponding processing load of a corresponding packet processor and ranges from a first load value, indicating a lowest processing load category, to an nth load value, indicating a highest processing load category, and

wherein "n" is an integer greater than one.

41. The method as claimed in claim 40, wherein said step (c) comprises:

(c1) evaluating the processing load values to determine said processing loads of said packet processors; and

5 (c2) selecting said packet processors categorized as having a
smallest processing load of said packet processors as said designated
processors to process said unclassified data packets.

42. The method as claimed in claim 35, wherein said step (c) comprises:

(c1) evaluating said load information and selecting said designated processors to process said unclassified data packets at frequencies that respectively correspond to said processing loads.

43. The method as claimed in claim 40, wherein said step (c) comprises:

(c1) evaluating said processing load values and selecting said designated processors to process said unclassified data packets at frequencies
5 that respectively correspond to said processing loads.

44. The method as claimed in claim 43, wherein said step (c1) comprises:

(c1a) grouping any of said packet processors that are associated with processing load values corresponding to a first processing load category of said processing load values into a first processor group and grouping any of said packet processors that are associated with processing load values that correspond to a second processing load category of said processing load values into a second processor group,

wherein said second processing load category represents a higher processing load category than said first processing load category; and

(c1b) selecting said packet processors in said first processor group to be some of said designated processors at a first frequency and selecting said packet processors in said second processor group to be some of said designated processors at a second frequency, wherein said first frequency is
15 greater than a second frequency.

45. The method as claimed in claim 44, wherein said first processing load category corresponds to a smallest processing load category of said packet processors and said second processing load category corresponds to a second smallest processing load category of said packet processors, and

5 wherein said step (c1b) comprises:

(c1b1) selecting at least some of said packet processors in said first processor group to be some of said designated processors during every round of selecting said designated processors; and

10 (c1b2) selecting at least some of said packet processors in said second processor group to be some of said designated processors during every two rounds of selecting said designated processors.

46. The method as claimed in claim 45,

wherein said step (c1a) comprises:

(c1a1) grouping any of said packet processors that are associated with processing load values corresponding to a third smallest processing load category into a third processor group and grouping any of said packet processors that are associated with processing load values
5

corresponding to a fourth smallest processing load category into a fourth processor group, and

wherein said step (c1b) further comprises:

10 (c1b3) selecting at least some of said packet processors in said third processor group to be some of said designated processors during every three rounds of selecting designated processors; and

(c1b2) selecting at least some of said packet processors in said fourth processor group to be some of said designated processors during every 15 four rounds of selecting designated processors.

47. The method as claimed in claim 40, wherein said step (c) comprises:

(c1) grouping any of said packet processors that are associated with said processing load values corresponding to said highest processing load 5 category in an unavailable processor group; and

(c2) refraining from selecting any of said packet processors in said unavailable group as said designated processors.

48. The method as claimed in claim 45,
wherein said step (c1a) comprises:

(c1a1) grouping any of said packet processors that are associated with said processing load values corresponding to said highest 5 processing load category into an unavailable group, and

wherein said step (c1b) further comprises:

(c1b3) refraining from selecting any of said packet processors in said unavailable group as said designated processors.

49. The method as claimed in claim 47, further comprising:

(d) when all of said packet processors are grouped in said unavailable group, resetting said processing load values respectively associated with said packet processors so that said processing load values 5 correspond to a processing load category that is less than said highest processing load category.

50. The method as claimed in claim 47, further comprising:

(d) when all of said packet processors are grouped in said unavailable group, reclassifying said processing load values respectively associated with said packet processors so that at least one of said processing 5 load values corresponds to a processing load category that is less than said highest processing load category.

51. The method as claimed in claim 47, further comprising:

(d) when all of said packet processors are grouped in said unavailable group, resetting said processing load values respectively associated with said packet processors so that said processing load values 5 correspond to said lowest processing load category.

52. The method as claimed in claim 48, further comprising:

(d) when all of said packet processors are grouped in said unavailable group, resetting said processing load values respectively associated with said packet processors so that said processing load values

5 correspond to a processing load category that is less than said highest processing load category.

53. The method as claimed in claim 48, further comprising:

(d) when all of said packet processors are grouped in said unavailable group, resetting said processing load values respectively associated with said packet processors so that said processing load values 5 correspond to said lowest processing load category.

54. The method as claimed in claim 35, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

wherein said packet processors respectively output said processing 5 load values to a controller, and

wherein said controller selects said designated processors based on said processing load values.

55. The method as claimed in claim 35, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

wherein a controller determines said processing load values of said 5 packet processors based on a percentage of a total number of process flows that said controller has assigned to each of said packet processors, and

wherein said controller selects said designated processors based on said processing load values.

56. The method as claimed in claim 35, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

wherein said processing load values respectively correspond to 5 amounts of process flows of data packets that have been respectively designated to be processed by said packet processors.

57. The method as claimed in claim 35, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors, and

wherein said processing load values respectively correspond idle times 5 of said packet processors.

58. The method as claimed in claim 35, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors, and

wherein said processing load values respectively correspond rates at 5 which data are supplied to said packet processors.

59. The method as claimed in claim 40, wherein said packet processors respectively output said processing load values to a controller, and

wherein said controller selects said designated processors based on said processing load values.

60. The method as claimed in claim 40, wherein a controller determines said processing load values of said packet processors based on a

percentage of a total number of process flows that said controller has assigned to each of said packet processors, and

5 wherein said controller selects said designated processors based on said processing load values.

61. The method as claimed in claim 40, wherein said processing load values respectively correspond to amounts of process flows of data packets that have been respectively designated to be processed by said packet processors.

62. The method as claimed in claim 40, wherein said processing load values respectively correspond idle times of said packet processors.

63. The method as claimed in claim 40, wherein said processing load values respectively correspond to rates at which data are supplied to said packet processors.

64. A software program contained in a computer readable medium for balancing a load among a plurality of packet processors, wherein the software program enables a controller to perform a routine, comprising:

5 (a) receiving input data packets from a stream of data packets;

 (b) determining which of said input data packets are unclassified data packets, wherein said unclassified data packets have unclassified data packet classifications and none of said packet processors has been previously designated to process data packets having any of said unclassified data packet classifications; and

10

(c) selecting designated processors of said packet processors to process said unclassified data packets based on load information, wherein said packet processors respectively have processing loads and said load information corresponds to said processing loads.

65. The software program as claimed in claim 64, wherein step (b) comprises:

(b1) determining which of said input data packets are classified data packets, wherein said classified data packets have classified data packet classifications and at least one of said packet processors has been previously designated to process data packets having said classified data packet classifications.

66. The software program as claimed in claim 65, wherein step (c) comprises:

(c1) supplying said classified data packets to said packet processors based on which of said packet processors have been previously designated to process data packets having said classified data packet classifications.

67. The software program as claimed in claim 64, wherein said unclassified data packet classifications correspond to process flows of said unclassified data packets.

68. The software program as claimed in claim 66, wherein said unclassified data packet classifications correspond to process flows of said unclassified data packets and wherein said classified data packet classifications correspond to process flows of said classified data packets.

69. The software program as claimed in claim 64, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

wherein each of said processing load values categorizes a corresponding processing load of a corresponding packet processor and ranges from a first load value, indicating a lowest processing load category, to an nth load value, indicating a highest processing load category, and

wherein "n" is an integer greater than one.

70. The software program as claimed in claim 69, wherein said step

(c) comprises:

(c1) evaluating the processing load values to determine said processing loads of said packet processors; and

5 (c2) selecting said packet processors categorized as having a smallest processing load of said packet processors as said designated processors to process said unclassified data packets.

71. The software program as claimed in claim 64, wherein said step

(c) comprises:

(c1) evaluating said load information and selecting said designated processors to process said unclassified data packets at frequencies that respectively correspond to said processing loads.

72. The software program as claimed in claim 69, wherein said step

(c) comprises:

(c1) evaluating said processing load values and selecting said designated processors to process said unclassified data packets at frequencies 5 that respectively correspond to said processing loads.

73. The software program as claimed in claim 72, wherein said step (c1) comprises:

(c1a) grouping any of said packet processors that are associated with processing load values corresponding to a first processing load category of 5 said processing load values into a first processor group and grouping any of said packet processors that are associated with processing load values that correspond to a second processing load category of said processing load values into a second processor group,

wherein said second processing load category represents a higher 10 processing load category than said first processing load category; and

(c1b) selecting said packet processors in said first processor group to be some of said designated processors at a first frequency and selecting said packet processors in said second processor group to be some of said designated processors at a second frequency, wherein said first frequency is 15 greater than a second frequency.

74. The software program as claimed in claim 73, wherein said first processing load category corresponds to a smallest processing load category of said packet processors and said second processing load category corresponds to a second smallest processing load category of said packet processors, and

5 wherein said step (c1b) comprises:

(c1b1) selecting at least some of said packet processors in said first processor group to be some of said designated processors during every round of selecting said designated processors; and

(c1b2) selecting at least some of said packet processors in said 10 second processor group to be some of said designated processors during every two rounds of selecting said designated processors.

75. The software program as claimed in claim 74,

wherein said step (c1a) comprises:

(c1a1) grouping any of said packet processors that are associated with processing load values corresponding to a third smallest 5 processing load category into a third processor group and grouping any of said packet processors that are associated with processing load values corresponding to a fourth smallest processing load category into a fourth processor group, and

wherein said step (c1b) further comprises:

10 (c1b3) selecting at least some of said packet processors in said third processor group to be some of said designated processors during every three rounds of selecting designated processors; and

(c1b2) selecting at least some of said packet processors in said 15 fourth processor group to be some of said designated processors during every four rounds of selecting designated processors.

76. The software program as claimed in claim 69, wherein said step (c) comprises:

(c1) grouping any of said packet processors that are associated with said processing load values corresponding to said highest processing load category in an unavailable processor group; and

(c2) refraining from selecting any of said packet processors in said unavailable group as said designated processors.

77. The software program as claimed in claim 74,
wherein said step (c1a) comprises:

(c1a1) grouping any of said packet processors that are associated with said processing load values corresponding to said highest processing load category into an unavailable group, and

wherein said step (c1b) further comprises:

(c1b3) refraining from selecting any of said packet processors in said unavailable group as said designated processors.

78. The software program as claimed in claim 76, further comprising:

(d) when all of said packet processors are grouped in said unavailable group, resetting said processing load values respectively associated with said packet processors so that said processing load values correspond to a processing load category that is less than said highest processing load category.

79. The software program as claimed in claim 76, further comprising:

5 (d) when all of said packet processors are grouped in said unavailable group, reclassifying said processing load values respectively associated with said packet processors so that at least one of said processing load values corresponds to a processing load category that is less than said highest processing load category.

80. The software program as claimed in claim 76, further comprising:

5 (d) when all of said packet processors are grouped in said unavailable group, resetting said processing load values respectively associated with said packet processors so that said processing load values correspond to said lowest processing load category.

81. The software program as claimed in claim 77, further comprising:

5 (d) when all of said packet processors are grouped in said unavailable group, resetting said processing load values respectively associated with said packet processors so that said processing load values correspond to a processing load category that is less than said highest processing load category.

82. The software program as claimed in claim 77, further comprising:

(d) when all of said packet processors are grouped in said unavailable group, resetting said processing load values respectively

5 associated with said packet processors so that said processing load values correspond to said lowest processing load category.

83. The software program as claimed in claim 64, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

5 wherein said packet processors respectively output said processing load values to a controller, and

wherein said controller selects said designated processors based on said processing load values.

84. The software program as claimed in claim 64, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

5 wherein a controller determines said processing load values of said packet processors based on a percentage of a total number of process flows that said controller has assigned to each of said packet processors, and

wherein said controller selects said designated processors based on said processing load values.

85. The software program as claimed in claim 64, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

5 wherein said processing load values respectively correspond to amounts of process flows of data packets that have been respectively designated to be processed by said packet processors.

86. The software program as claimed in claim 64, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors, and wherein said processing load values respectively correspond idle times 5 of said packet processors.

87. The software program as claimed in claim 64, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors, and wherein said processing load values respectively correspond rates at 5 which data are supplied to said packet processors.

88. The software program as claimed in claim 69, wherein said packet processors respectively output said processing load values to a controller, and wherein said controller selects said designated processors based on 5 said processing load values.

89. The software program as claimed in claim 69, wherein a controller determines said processing load values of said packet processors based on a percentage of a total number of process flows that said controller has assigned to each of said packet processors, and 5 wherein said controller selects said designated processors based on said processing load values.

90. The software program as claimed in claim 69, wherein said processing load values respectively correspond to amounts of process flows of

data packets that have been respectively designated to be processed by said packet processors.

91. The software program as claimed in claim 69, wherein said processing load values respectively correspond idle times of said packet processors.

92. The software program as claimed in claim 69, wherein said processing load values respectively correspond to rates at which data are supplied to said packet processors.

93. An apparatus, comprising:
a plurality of packet processors; and
a controller, wherein said controller receives input data packets from a stream of data packets and determines which of said input data packets are 5 unclassified data packets,

wherein said unclassified data packets have unclassified data packet classifications and none of said packet processors has been previously designated to process data packets having any of said unclassified data packet classifications,

10 wherein said controller selects designated processors to process said unclassified data packets based on load information, and

wherein said packet processors respectively have processing loads and said load information corresponds to said processing loads.

94. The apparatus as claimed in claim 93, wherein said controller determines which of said input data packets are classified data packets,

wherein said classified data packets have classified data packet classifications and at least one of said packet processors has been previously designated to 5 process data packets having said classified data packet classifications.

95. The apparatus as claimed in claim 94, wherein said controller supplies said classified data packets to said packet processors based on which of said packet processors have been previously designated to process data packets having said classified data packet classifications.

96. The apparatus as claimed in claim 93, wherein said unclassified data packet classifications correspond to process flows of said unclassified data packets.

97. The apparatus as claimed in claim 95, wherein said unclassified data packet classifications correspond to process flows of said unclassified data packets and wherein said classified data packet classifications correspond to process flows of said classified data packets.

98. The apparatus as claimed in claim 93, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

wherein each of said processing load values categorizes a 5 corresponding processing load of a corresponding packet processor and ranges from a first load value, indicating a lowest processing load category, to an nth load value, indicating a highest processing load category, and

wherein "n" is an integer greater than one.

99. The apparatus as claimed in claim 98, wherein said controller evaluates the processing load values to determine said processing loads of said packet processors, and selects said packet processors categorized as having a smallest processing load of said packet processors as said designated 5 processors to process said unclassified data packets.

100. The apparatus as claimed in claim 93, wherein said controller evaluates said load information and selects said designated processors to process said unclassified data packets at frequencies that respectively correspond to said processing loads.

101. The apparatus as claimed in claim 98, wherein said controller evaluates said processing load values and selects said designated processors to process said unclassified data packets at frequencies that respectively correspond to said processing loads.

102. The apparatus as claimed in claim 101, wherein said controller groups any of said packet processors that are associated with processing load values corresponding to a first processing load category of said processing load values into a first processor group and groups any of said packet 5 processors that are associated with processing load values that correspond to a second processing load category of said processing load values into a second processor group,

wherein said second processing load category represents a higher processing load category than said first processing load category,

10

wherein said controller selects said packet processors in said first processor group to be some of said designated processors at a first frequency and selects said packet processors in said second processor group to be some of said designated processors at a second frequency, and
wherein said first frequency is greater than a second frequency.

103. The apparatus as claimed in claim 102, wherein said first processing load category corresponds to a smallest processing load category of said packet processors and said second processing load category corresponds to a second smallest processing load category of said packet processors, and

5

wherein said controller selects at least some of said packet processors in said first processor group to be some of said designated processors during every round of selecting said designated processors and selects at least some of said packet processors in said second processor group to be some of said designated processors during every two rounds of selecting said designated

10

processors.

5

104. The apparatus as claimed in claim 103, wherein said controller groups any of said packet processors that are associated with processing load values corresponding to a third smallest processing load category into a third processor group and groups any of said packet processors that are associated with processing load values corresponding to a fourth smallest processing load category into a fourth processor group,

wherein said controller selects at least some of said packet processors in said third processor group to be some of said designated processors during every three rounds of selecting designated processors,

10 wherein said controller selects at least some of said packet processors in said fourth processor group to be some of said designated processors during every four rounds of selecting designated processors.

105. The apparatus as claimed in claim 98, wherein said controller groups any of said packet processors that are associated with said processing load values corresponding to said highest processing load category in an unavailable processor group, and

5 wherein said controller refrains from selecting any of said packet processors in said unavailable group as said designated processors.

106. The apparatus as claimed in claim 103, wherein said controller groups any of said packet processors that are associated with said processing load values corresponding to said highest processing load category into an unavailable group, and

5 wherein said controller refrains from selecting any of said packet processors in said unavailable group as said designated processors.

107. The apparatus as claimed in claim 105, wherein, when all of said packet processors are grouped in said unavailable group, said controller resets said processing load values respectively associated with said packet processors so that said processing load values correspond to a processing load category that is less than said highest processing load category.

108. The apparatus as claimed in claim 105, wherein, when all of
said packet processors are grouped in said unavailable group, said controller
resets said processing load values respectively associated with said packet
processors so that said processing load values correspond to said lowest
5 processing load category.

109. The apparatus as claimed in claim 105, wherein, when all of
said packet processors are grouped in said unavailable group, said controller
reclassifies said processing load values respectively associated with said
packet processors so that at least one of said processing load values
5 corresponds to a processing load category that is less than said highest
processing load category.

110. The apparatus as claimed in claim 106, wherein, when all of
said packet processors are grouped in said unavailable group, said controller
resets said processing load values respectively associated with said packet
processors so that said processing load values correspond to a processing load
5 category that is less than said highest processing load category.

111. The apparatus as claimed in claim 106, wherein, when all of
said packet processors are grouped in said unavailable group, said controller
resets said processing load values respectively associated with said packet
processors so that said processing load values correspond to said lowest
5 processing load category.

112. The apparatus as claimed in claim 93, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

wherein said packet processors respectively output said processing
5 load values to said controller, and

wherein said controller selects said designated processors based on said processing load values.

113. The apparatus as claimed in claim 93, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

wherein said controller determines said processing load values of said
5 packet processors based on a percentage of a total number of process flows that said controller has assigned to each of said packet processors, and

wherein said controller selects said designated processors based on said processing load values.

114. The apparatus as claimed in claim 93, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors,

wherein said processing load values respectively correspond to
5 amounts of process flows of data packets that have been respectively designated to be processed by said packet processors.

115. The apparatus as claimed in claim 93, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors, and

wherein said processing load values respectively correspond idle times
5 of said packet processors.

116. The apparatus as claimed in claim 93, wherein said load information comprises processing load values that respectively correspond to said processing loads of said packet processors, and

wherein said processing load values respectively correspond rates at
5 which data are supplied to said packet processors.

117. The apparatus as claimed in claim 98, wherein said packet processors respectively output said processing load values to said controller, and

wherein said controller selects said designated processors based on
5 said processing load values.

118. The apparatus as claimed in claim 98, wherein said controller determines said processing load values of said packet processors based on a percentage of a total number of process flows that said controller has assigned to each of said packet processors, and

5 wherein said controller selects said designated processors based on said processing load values.

119. The apparatus as claimed in claim 98, wherein said processing load values respectively correspond to amounts of process flows of data

packets that have been respectively designated to be processed by said packet processors.

120. The apparatus as claimed in claim 98, wherein said processing load values respectively correspond idle times of said packet processors.

121. The apparatus as claimed in claim 98, wherein said processing load values respectively correspond to rates at which data are supplied to said packet processors.